APPENDIX

9. (Amended) A computing system comprising:

a compiler for forming groups of instructions having opcodes including a first group of instructions and a second group of instructions, instructions in the first group of instructions executable in parallel, and instructions in the second group of instructions executable in parallel;

a first memory storage having at least a memory location, the memory location for storing the first group of instructions, for storing the second group of instructions comprising at least one instruction, and for storing group identifiers that indicate which instructions are included within the first group of instructions and which instructions are included within the second group of instructions;

a pre-decoder coupled to the first memory storage for decoding opcodes of instructions in the first group of instructions and opcodes of instructions in the second group of instructions, for forming a first group of expanded instructions, a second group of expanded instructions, and expanded group identifiers, and for determining processing pipeline identifiers associated with expanded instructions in the first group of expanded instructions and processing pipeline identifiers associated with expanded instructions in the second group of expanded instructions, in response thereto;

a second memory storage coupled to the predecoder having at least a memory location, the memory location for storing the first group of expanded instructions, the second group of expanded instructions, the expanded group identifiers, the processing pipeline identifiers associated with the expanded instructions in the first group of expanded instructions, and the processing pipeline identifiers associated with the expanded instructions in the second group of expanded instructions;

a decoder coupled to the second memory storage for receiving the first group of expanded instructions, the second group of expanded instructions, and the expanded group identifiers, and for issuing the first group of expanded instructions in response to the expanded group identifiers;

a plurality of processing pipelines;

a crossbar coupled to the decoder and to the plurality of processing pipelines, for issuing expanded instructions in the first group of expanded instructions to processing pipelines of the plurality of processing pipelines in response to the processing pipeline identifiers associated with the expanded instructions in the first group of expanded instructions.

10. The computing system of claim 9 wherein a first processing pipeline identifier from the processing pipeline identifiers associated with the expanded instructions in the first group of expanded instructions identifies a first floating point unit pipeline.

11. The computing system of claim 10 wherein a second processing pipeline identifier from the processing pipeline identifiers associated with the expanded

instructions in the first group of expanded instructions identifies a first arithmetic logic unit pipeline.

12. The computing system of claim 10 wherein a second processing pipeline identifier from the processing pipeline identifiers associated with the expanded instructions in the first group of expanded instructions identifies a second floating point unit pipeline.

13. The computing system of claim 11 wherein a third processing pipeline identifier from the processing pipeline identifiers associated with the expanded instructions in the first group of expanded instructions identifies a second floating point unit pipeline.

14. The computing system of claim 9 wherein a first processing pipeline identifier from the processing pipeline identifiers associated with the expanded instructions in the second group of expanded instructions identifies a store unit pipeline.

15. The computing system of claim 14 wherein a second processing pipeline identifier from the processing pipeline identifiers associated with the expanded instructions in the second group of expanded instructions identifies a control unit pipeline.

16. The computing system of claim 9

wherein the decoder is also for issuing the second group of expanded instructions in response to the expanded group identifiers; and

wherein the crossbar is also for issuing expanded instructions in the second group of expanded instructions to processing pipelines of the plurality of processing pipelines in response to the processing pipeline identifiers associated with the expanded instructions in the second group of expanded instructions.

17. The computing system of claim 9 wherein the group identifiers are associated with instructions in the first group of instructions and with instructions in the second group of instruction.

18. The computing system of claim 17 wherein the group identifiers are embedded with instructions in the first group of instructions and with instructions in the second group of instruction.

19. The computing system of claim 9 wherein the first group of expanded instructions comprises at least one expanded instruction and the second group of expanded instructions comprises at least two expanded instructions.

20. The computing system of claim 19 wherein the second memory storage includes at least the two expanded instructions of the second group of expanded instructions and the one expanded instruction of the first group of expanded instructions.

21. The computing system of claim 9

wherein the compiler is also for forming a third group of instructions, the instructions in the third group of instructions executable in parallel;

wherein the memory location of the first memory storage is also for storing a third group of instructions in parallel with the first group of instructions and the second group of instructions; and

wherein the group identifiers indicate which instructions are included within the third group of instructions.

22. The computing system of claim 21 wherein the third group of instructions comprises at least one instruction.

23. The computing system of claim 9 wherein the first memory storage is a superscaler cache.

24. The computing system of claim 9 wherein one expanded instruction of the second group of expanded instructions is a branch instruction.

25. The computing system of claim 9 wherein the compiler explicitly identifies instructions that can be performed in parallel for the first group of instructions.

26. (Amended) A method for issuing groups of individual software-scheduled instructions in parallel for processing comprises:

forming a first group of software-scheduled instructions, a second group of software-scheduled instructions comprising at least one instruction, and group identifiers indicating which software-scheduled instructions are included within the first group of software-scheduled instructions and which software-scheduled instructions are included within the second group of software-scheduled instructions, software-scheduled instructions in the first group of software-scheduled instructions having opcodes and executable in parallel, and software-scheduled instructions in the second group of software-scheduled instructions having opcodes and executable in parallel;

storing the first group of software-scheduled instructions, the second group of software-scheduled instructions, and the group identifiers in parallel in a first memory location;

forming a first group of expanded software-scheduled instructions, a second group of expanded software-scheduled instructions, and expanded group identifiers in response to opcodes of software-scheduled instructions in the first group of software-scheduled instructions and opcodes of software-scheduled instructions in the second group of software-scheduled instructions;

determining processing pipelines appropriate for expanded software-scheduled instructions in the first group of expanded software-scheduled instructions and processing pipelines appropriate for expanded software-scheduled instructions in the second group of expanded software-scheduled instructions also in response to the opcodes of software-scheduled instructions in the first group of software-scheduled instructions and the opcodes

of software-scheduled instructions in the second group of software-scheduled instructions; and

issuing the first group of expanded software-scheduled instructions to the processing pipelines appropriate for expanded software-scheduled instructions in the first group of expanded software-scheduled instructions, in response to the expanded group identifiers.

27. The method of claim 26 further comprising:

issuing the second group of expanded software-scheduled instructions to the processing pipelines appropriate for expanded software-scheduled instructions in the second group of expanded software-scheduled instructions, in response to the expanded group identifiers.

28. The method of claim 26 wherein a first processing pipeline appropriate for an expanded software-scheduled instruction in the first group of expanded software-scheduled instructions is coupled to a first arithmetic logic processing unit.

29. The method of claim 28 wherein a second processing pipeline appropriate for an expanded software-scheduled instruction in the first group of expanded software-scheduled instructions is coupled to a first floating point processing unit.

30. The method of claim 28 wherein a second processing pipeline appropriate for an expanded software-scheduled instruction in the first group of expanded software-scheduled instructions is coupled to a second arithmetic logic processing unit.

31. The method of claim 29 wherein a third processing pipeline appropriate for an expanded software-scheduled instruction in the first group of expanded software-scheduled instructions is coupled to a second arithmetic logic processing unit.

32. The method of claim 28 wherein a second processing pipeline appropriate for an expanded software-scheduled instruction in the first group of expanded software-scheduled instructions is coupled to a load unit.

33. The method of claim 26 wherein a first processing pipeline appropriate for an expanded software-scheduled instruction in the second group of expanded software-scheduled instructions is coupled to a store unit.

34. The method of claim 33 wherein a second processing pipeline appropriate for an expanded software-scheduled instruction in the second group of expanded software-scheduled instructions is coupled to a control unit.

35. The method of claim 26 further comprises:

storing the first group of expanded software-scheduled instructions, the second group of expanded software-scheduled instructions, and the expanded group identifiers, in a second memory location;

wherein the step of issuing the first group of expanded software-scheduled instructions comprises issuing the first group of expanded software-scheduled instructions and the second group of expanded software-scheduled instructions from the second memory location to a decoder; and

issuing the first group of expanded software-scheduled instructions to the processing pipelines appropriate for the expanded software-scheduled instructions in the first group of expanded software-scheduled instructions from the decoder in response to the expanded group identifiers stored in the second memory location.

36. The method of claim 35 further comprises:

issuing the second group of expanded software-scheduled instructions to the processing pipelines appropriate for the expanded software-scheduled instructions in the second group of expanded software-scheduled instructions from the decoder in response to the expanded group identifiers stored in the second memory location.

37. The method of claim 36 wherein the processing pipelines appropriate for the expanded software-scheduled instructions in the first group of expanded software-scheduled instructions are identified by processing pipeline identifiers.

38. The method of claim 37 wherein the step of storing the first group of expanded software-scheduled instructions further comprises storing the processing pipeline identifiers in the second memory location.

39. The method of claim 38 wherein the step of issuing the first group of expanded software-scheduled instructions to the processing pipelines comprises issuing the first group of expanded software-scheduled instructions to a crossbar.

40. The method of claim 39 wherein the crossbar is an associative crossbar responsive to the processing pipeline identifiers.

41. The method of claim 35 wherein the first group of expanded instructions comprises at least two expanded instructions.

42. The method of claim 41 wherein the second group of expanded instructions comprises at least one expanded instruction.

43. The method of claim 42 wherein the two expanded instructions of the first group of expanded instructions and the one expanded instruction of the second group of expanded instructions are stored in the second memory location

44. The method of claim 26 wherein the step of forming the first group

of software-scheduled instructions, the second group of software-scheduled instructions, and the group identifiers comprises using a compiler to form the first group of software-scheduled instructions, the second group of software-scheduled instructions, and the group identifiers.

45. The method of claim 44 wherein the group identifiers are associated with instructions in the first group of instructions and with instructions in the second group of instruction.

46. The method of claim 44 wherein the compiler explicitly determines parallel executable instructions among a plurality of instructions to include in the first group of software-scheduled instructions.

47. The method of claim 26 wherein the step of forming the first group of software-scheduled instructions, the second group of software-scheduled instructions, and the group identifiers further comprises the step of forming a third group of software-scheduled instructions executable in parallel; and

wherein the group identifiers also indicate which software-scheduled instructions are included within the third group of software-scheduled instructions.

48. The method of claim 26 wherein the first memory location is a cache location in a superscaler cache.

49. (Amended) A method for issuing a group of individual instructions in parallel for processing comprises:

storing in parallel a plurality of instructions and instruction grouping information in a location in a memory, the plurality of instructions and the instruction grouping information determined by a compiler, the instruction grouping information indicating which instructions of the plurality of instructions belong to a first group of instructions and can be issued in parallel, and indicating at least another instruction of the plurality of instructions that can be issued after the first group of instructions;

issuing the first group of instructions in response to the instruction grouping information; and

coupling instructions in the first group of instructions to instruction pipelines appropriate for the instructions in the first group of instructions.

50. (Amended) The method of claim 49 further comprises

after coupling instructions in the first group of instructions, issuing the at least another instruction in response to the instruction grouping information; and

coupling the at least another instruction to an instruction pipeline appropriate for the at least another instruction .

51. (Amended) The method of claim 50 wherein the instruction grouping information also indicates which instructions of the plurality of instructions belong to a

second group of instructions and can be issued in parallel after the at least another instruction .

52. (Amended) The method of claim 49
wherein the instructions in the first group of instructions include instruction types; and
wherein coupling instructions in the first group of instructions further comprises determining the instruction pipelines appropriate for the instructions in the first group of instructions in response to the instruction types.

53. The method of claim 52 wherein the instruction types comprise opcodes.

54. (Amended) The method of claim 50 wherein issuing the first group of instructions further comprises receiving the first group of instructions, the at least another instruction , and the instruction grouping information from the location in the memory.

55. (Amended) The method of claim 50 wherein the first group of instructions comprises at least two instructions.

56. (Amended) The method of claim 50 wherein the first group of instructions comprises at least one instruction.

57. (Amended) The method of claim 55 wherein an instruction frame comprises the plurality of instructions and instruction grouping information, and
wherein the instruction frame includes at least the two instructions of the first group of instructions and the at least another instruction.

58. The method of claim 50 wherein the compiler explicitly identifies parallel executable instructions from the plurality of instructions.

59. The method of claim 52 wherein the instruction types comprise pipeline identifiers that identify the instruction pipelines appropriate for the instructions in the first group of instructions.

60. The method of claim 59 wherein the pipeline identifiers are determined by the compiler.

61. (Amended) The method of claim 60 wherein coupling instructions in the first group of instructions comprises using a crossbar switch to couple the instructions in the first group of instructions to the instruction pipelines appropriate for the instructions in the first group of instructions in response to the pipeline identifiers.

62. The method of claim 50 wherein the memory is a cache and the location is a cache entry.

63. (Amended) A computing system comprising:

a cache including a plurality of cache entries, a cache entry of the plurality of cache entries configured to store in parallel a plurality of software-scheduled instructions and instruction grouping information, the instruction grouping information configured to identify a first group of software-scheduled instructions from the plurality of software-scheduled instructions and to identify at least another software-scheduled instruction from the plurality of software-scheduled instructions, the at least another software-scheduled instruction to be issued after instructions in the first group of software-scheduled instructions.

64. (Amended) The computing system of claim 63 wherein the instruction grouping information is also configured to identify a second group of software-scheduled instructions from the plurality of software-scheduled instructions, instructions in the second group of software-scheduled instructions to be issued after at least another software-scheduled instruction.

65. (Amended) The computing system of claim 63

wherein the cache is also configured to issue the first group of software-scheduled instructions, the at least another software-scheduled instruction, and the instruction grouping information;

the computing system further comprising a group decoder coupled to the cache and configured to receive the first group of software-scheduled instructions, the at least another software-scheduled instruction, and the instruction grouping information, and to issue the first group of software-scheduled instructions in response to the instruction grouping information.

66. (Amended) The computing system of claim 65 wherein the group decoder is also configured to issue the at least another software-scheduled instruction, after the first group of software-scheduled instructions in response to the instruction grouping information.

67. (Amended) The computing system of claim 65

wherein each instruction in the first group of software-scheduled instructions includes an instruction type,

the computing system further comprising an instruction decoder coupled to the cache and configured to receive the instruction types of the instructions in the first group of software-scheduled instructions and to determine instruction pipelines appropriate for each of the instructions in the first group of software-scheduled instructions.

68. The computing system of claim 67 wherein the instruction type comprises an opcode, and the instruction decoder comprises an opcode decoder.

69. The computing system of claim 67 further comprising:

a pipeline coupler coupled to the group decoder and to the instruction decoder and configured to receive the first group of software-scheduled instructions and configured to couple each instruction in the first group of software-scheduled instructions to the instruction pipelines appropriate for the instructions in the first group of software-scheduled instructions.

70. The computing system of claim 69 wherein the pipeline coupler is a crossbar switch.

71. The computing system of claim 65 wherein the first group of software-scheduled instructions comprises at least two instructions.

72. (Amended) The computing system of claim 71 wherein the first group of software-scheduled instructions comprises at least one instruction.

73. (Amended) The computing system of claim 72 wherein an instruction frame comprises the plurality of software-scheduled instructions and instruction grouping information; and wherein the instruction frame includes at least the two instructions of the first group of software-schedule instructions and the at least another software-scheduled instruction.

74. (Amended) The computing system of claim 63 wherein the cache is a superscaler cache.

75. The computing system of claim 63 wherein a compiler explicitly identifies parallel executable instructions from the plurality of software-scheduled instructions that form the first group of software-scheduled instructions.

76. (Amended) The computing system of claim 67 wherein the instruction types comprise pipeline identifiers indicative of instruction pipelines appropriate for the instructions in the first group of software-scheduled instructions.

77. (New) The method of claim 49 wherein the instruction grouping information also indicates instruction pipelines appropriate for the instructions in the first group of instructions; and

wherein coupling instructions in the first group of instructions to the instruction pipelines appropriate for the instructions in the first group of instructions is in response to the instruction grouping information.

78. (New) The method of claim 77 wherein the instruction grouping information also indicates instruction pipelines appropriate for the instructions in the second group of instructions.

79. (New) The computing system of claim 65 further comprising an switching unit coupled to the cache and configured to receive the instructions in the first group of software-scheduled instructions and to couple the instructions in the first group of software-scheduled instructions to instruction pipelines appropriate for each of the instructions in the first group of software-scheduled instructions in response to the instruction grouping information.

80. (New) The computing system of claim 79 wherein the switching unit is also configured to receive the at least another software-scheduled instruction and to couple the at least another software-scheduled instruction to an instruction pipeline appropriate the at least another software-scheduled instruction in response to the instruction grouping information.

81. (New) A computing system in which instructions are issued in parallel to processing pipelines, the computing system comprising:

a storage configured to store an instruction frame, the instruction frame including a plurality of instructions including a group of instructions and at least another instruction in parallel, instructions in the group of instructions to be issued in parallel, the instruction frame also including data associated with the plurality of instructions, the data associated with the plurality of instructions indicative of which instructions in the plurality of instructions are included in the group of instructions, the data associated with the plurality of instructions also indicative of processing pipelines appropriate for the plurality of instructions, and the data associated with the plurality of instructions determined at a compile time; and

a switching circuit coupled to the storage, configured to issue the instructions in the group of instructions in parallel, to processing pipelines appropriate for the instructions in the group of instructions, in response to the data associated with the plurality of instructions.

82. (New) The computing system of claim 81 wherein the group of instructions comprises at least two instructions.

83. (New) The computing system of claim 81 wherein the group of instructions comprises one instruction.

84. (New) The computing system of claim 81 wherein the switching circuit is also configured to issue the at least another instruction to a processing pipeline appropriate for the at least another instruction in response to the data associated with the plurality of instructions.

85. (New) The computing system of claim 81 wherein the processing pipelines appropriate for the instructions in the group of instructions are respectively coupled to execution units appropriate for the instructions in the group of instructions.

86. (New) The computing system of claim 85 wherein an execution unit appropriate for a first instruction in the group of instructions is a memory.

87. (New) The computing system of claim 86 wherein an execution unit appropriate for a second instruction in the group of instructions is an arithmetic logic unit.

88. (New) The computing system of claim 86 wherein an execution unit appropriate for a second instruction in the group of instructions is a floating point unit.

89. (New) The computing system of claim 85 wherein an execution unit appropriate for one instruction in the group of instructions is a branch unit.

90. (New) The computing system of claim 85 wherein a type of execution unit appropriate for a first instruction in the group of instructions and a type of execution unit appropriate for a second instruction in the one group of instructions are similar.

91. (New) The computing system of claim 81 wherein the plurality of instructions in the instruction frame are determined at the compile time.

92. (New) A method for issuing groups of instructions in parallel to processing pipelines, the method comprising:

storing in a storage, an instruction frame, the instruction frame including a plurality of instructions including a group of instructions and at least another instruction in parallel, instructions in the group of instructions to be issued in parallel, the instruction frame also including data fields associated with the plurality of instructions, the data associated with the plurality of instructions indicative of which instructions are included in the group of instructions, the data associated with the instructions also indicative of processing pipelines appropriate for the plurality of instructions, and the data associated with the plurality of instructions determined at compile time; and

issuing the instructions in the group of instructions in parallel to processing pipelines appropriate for the instructions in the group of instructions, in response to the data associated with the plurality of instructions.

93. (New) The method of claim 92 further comprising:

during compile time, determining the plurality of instructions in the instruction frame.

94. (New) The method of claim 92 wherein the group of instructions comprises at least two instructions.

95. (New) The method of claim 94 further comprising, before issuing the group of instructions, issuing the at least another instruction to a processing pipeline appropriate for the at least another instruction in response to the data associated with the plurality of instructions.

96. (New) The method of claim 92 wherein the processing pipelines appropriate for the instructions in the group of instructions are respectively coupled to execution units appropriate for the instructions in the group of instructions.

97. (New) The method of claim 96 wherein a type of execution unit appropriate for a processing pipeline appropriate for a first instruction in the group of instructions is an arithmetic logic unit.

98. (New) The method of claim 97 wherein a type of execution unit appropriate for a processing pipeline appropriate for a second instruction in the group of instructions is an arithmetic logic unit.

99. (New) The method of claim 96 wherein a type of execution unit appropriate for a processing pipeline appropriate for a first instruction in the group of instructions is a floating point unit.

100. (New) The method of claim 96 wherein a type of execution unit appropriate for a processing pipeline appropriate for a first instruction in the group of instructions is a memory unit and a type of execution unit appropriate for a processing pipeline appropriate for a second instruction in the group of instructions is a memory unit.

101. (New) A method of operating a microprocessor comprises:

compiling computer code to determine a frame of instructions;

storing in a memory storage the frame of instructions, the frame of instructions including a plurality of instructions and issue data, the plurality of instructions including at least a first instruction, a second instruction, and a third instruction, the issue data comprising data indicating that the first instruction is to be issued before the second instruction and the third instruction and the second and third instructions are to be issued in parallel, and the issue data indicating respective processing units appropriate for the first instruction, the second instruction, and the third instruction; and

issuing the first instruction to a processing unit appropriate for the first instruction in response to the issue data; and

issuing the second instruction and the third instruction in parallel to respective processing units appropriate for the second instruction and the third instruction in response to the issue data.

102. (New) The method of claim 101 wherein the processing unit appropriate for the first instruction is a memory unit.

103. (New) The method of claim 102 wherein a processing unit appropriate for the second instruction is a memory unit.

104. (New) The method of claim 102 wherein a processing unit appropriate for the second instruction is an arithmetic logic unit.

105. (New) The method of claim 101 wherein issuing the first instruction to a processing unit comprises using a switching unit to couple the first instruction to the processing unit appropriate for the first instruction in response to the issue data.